Next AUG Meeting
Sunday, February 19th, 1989 at 2pm

(Doors open at 1pm, meeting starts at 2pm sharp)

**AUG meetings are held at Victoria College Burwod Campus
Burwood Highway, Burwood Melways map 61 reference B5.**

Amiga Users Group Inc, PO Box 48, Boronia, 3155, Victoria, Australia

AMIGA™ Users Group

Who Are WE?

The Amiga Users Group is a not-for-profit association of people interested in the Amiga computer and related topics. With over 1000 members, we are the largest independent association of Amiga users in Australia.

Club Meetings

Club meetings are held at 2pm on the third Sunday of each month at Victoria College, Burwood Highway, Burwood. Details on how to get there are on the back cover of this newsletter. The dates of upcoming meetings are:

Sunday, February 19th at 2pm

Sunday, March 19th at 2pm

Sunday, April 16th at 2pm

Production Credits

This month's newsletter was edited by Con Kolivas. Equipment and software used was: Amiga 500, Excellence!, Iff2ps and Apple LaserWriter Plus.

Copyright and Reprint Privileges

Amiga Workbench is copyright 1988 by the Amiga Users Group Inc. Articles herein that are copyrighted by individual authors or otherwise explicitly marked as having restricted reproduction rights may not be reprinted or copied without written permission from the Amiga Users Group or the authors. All other articles may be reprinted for any non-commercial purpose if accompanied by a credit line including the original author's name and the words "Reprinted from Amiga Workbench, newsletter of the Amiga Users Group, PO box 48, Boronia, 3155".

Contributions

Articles, papers, letters, drawings, cartoons and comments are actively sought for publication in Amiga Workbench. All contributions submitted for the purpose of publication that are printed in the newsletter are rewarded on the basis of one free public domain disk copy per column or half page printed with a minimum of one free copy. Contributions may be sent in on disk, paper or uploaded to Amiga Link or Amiga Link II in the area set aside for this purpose. Please send your contributions in text-only, non-formatted if they are on file and remember to include your address for return of disks and tokens for PD disks. Absolute deadline for articles is 16 days before the meeting date. Contributions can be sent to: The Editor, AUG, PO box 48, Boronia, 3155.

Membership and Subscriptions

Membership of the Amiga Users Group is available for an annual fee of $25. To become a member of AUG, fill in the membership form in this issue (or a photocopy of it), and send it with a cheque or money order for $25 to: Amiga Users Group, PO Box 48, Boronia, 3155

Public Domain Software

Disks from our public domain library are available on quality 3.5" disks for $8 each including postage on AUG supplied disks, or $2 each on your own disks. The group currently holds over 200 volumes, mostly sourced from the USA, with more on the way each month. Details of latest releases are printed in this newsletter, and a catalog disk is also available.

Member's Discounts

The Amiga Users Group negotiates discounts for its members on hardware, software and books.

Currently, **Technical Books** in Swanston Street in the city offers AUG members a 10% discount on computer related books, as does McGills in Elizabeth Street. Just show your membership card. Although we have no formal arrangements with other companies yet, most seem willing to offer a discount to AUG members. It always pays to ask!

Back Issues of Workbench

All back issues of Amiga Workbench are now available, for $2 each including postage. Note that there may be delays while issues are reprinted. Back issues are also available at meetings.

Amiga Link I & II - Our Bulletin Board Systems

The Amiga Users Group operates two bulletin board systems devoted to the Amiga, using the Opus message and conferencing software. AmigaLink I and II are available 24 hours a day. Both AmigaLink I & 2 can be accessed at V21 (300bps), V22 (1200bps), V23 (1200/75bps) and AmigaLink I also accepts V22bis (2400bps) using 8 data bits, 1 stop bit and no parity. Phone numbers for Amigalink are:

Amigalink I - (03) 792 3918
Amigalink II - (03) 376 6385

AmigaLink is part of a world-wide network of bulletin boards, and we participate in national and international Amiga conferences. AmigaLink has selected Public Domain software available for downloading, and encourages the uploading of useful public domain programs from its users. AmigaLink I is OzNet node number 8:830/324 and AmigaLink II is OzNet node number 1305/998

Newsletter Advertising

The Amiga Users Group accepts commercial advertising in Amiga Workbench subject to the availability of space at these rates:

Quarter page	$20
Half page	$40
Full page	$70

These rates are for full-size camera-ready copy only. We have no photographic or typesetting facilities. Absolute deadline for copy is 16 days before the meeting date. Send the copy and your cheque to: The Editor, AUG, PO Box 48, Boronia, 3155, Victoria.

Amiga Users Group Committee

Co-ordinator:	Bob Scarfe	882 1445	Hawthorn
Vice Co-ord:	Lester McClure	233 5664	Mt. Waverley
Meeting Chair:	Ron Wail	848 7283	Doncaster
Secretary:	John Elston	375 4142	Moonee Ponds
Treasurer:	Donna Heenan	729 2327	Bayswater
Membership:	Neil Murray	794 5683	Dandenong
Purchasing:	Bohdan Ferens	792 1138	Dandenong
Book Library:	Ross Johnson	824 7026	Malvern
Disk Library:	Craig Hutchinson		Carnegie
Editor:	Con Kolivas	484 1339	Thornbury
Committee:	Doug Myers	428 0682	Abbotsford
	Peter Jetson	752 2053	

MachII 2.2a
by Lu Beranek

With everybody writing about their workbenches and startup sequences and obviously nobody has the perfect setup except the writer of that particular article, I've decided to hop in for my chop.

First in line is MachII. First because it's one of those program's that if it disappears for any length of time, you start to get withdrawal symptoms (fortunately it doesn't disappear). It's a 'donations accepted'-ware program. Being of the opinion that what I prefer to pay for is the program (after extensive testing), I sent off a donation to Brian Moats and was pleasantly surprised to receive, a mere 2 weeks later, a disc with the latest version of Mach.

And now to the program itself. Not just another clock-in-the-title-bar utility. No, no, no, no! This one has a wealth of extras. After it is up if you shift the pointer over the clock/memory window and click the right mouse button, a full length window appears showing a whole range of options.

Starting from the top, there are 20 user definable macros or hotkeys based on Amiga and Ctrl-Amiga combined with the function keys. The macros are limited to 255 characters by default but you can change that to 9999 characters for that killer macro. Then there is the string input box for a popcli type function. This means that pressing a left Amiga/Escape combination will give you a Newcli or NewShell according to your preferences. A SunMouse type option - i.e. a window becomes active when a pointer is moved into it - is handy. Further window functions are 1 or 2 left mouse button clicks to bring windows to the front and left Amiga/M to shuffle screens in all cases.

There is a qualifier accumulator. This gadget means that for those people who have trouble pressing several buttons simultaneously can now press them one at time and have the keystrokes accumulate until they press the return key. Screen blanking can be set between 0-99 minutes, a mouse accelerator on it's highest setting allows your cursor to flick from side to side of the screen with the slightest movement. This facility also doubles as a detector of low temperatures or DT's (I won't go into how one tells the difference).

A low memory warning and user definable key combinations to allow matching of the first occurrence of a macro to a string segment and macro learn and execute are functions which I've yet to use, but one day, real soon now...

Oh yes, the clock. It can show the time, date, free chip and fast RAM, auto to the front of any window, be moved or defined in any position, give beeps at regular intervals, indicate dollars if you are measuring online costs, has a 24 hour alarm and most importantly, can be turned off completely.

If that is not enough to handle your many and varied needs, then you can also save different applications in configuration files to suit.

===

FAERY HINTS
by Rimon Abohaidar.

Here's just a few more hints to make life easier for Faery Talers.

TOTEMS- A good way to save totems is to save your current game, use a totem to see where you are and then reload the game you just saved.

SHELL- The shell is in the Seahold. You need a grey key.

RINGS- A good way to use a ring if you have plenty of them is to save them until you are really out-numbered (5 goblins, etc.) use a ring and then stand in front of one them stabbing your sword at him. Since you can't hurt them while the ring is working, as soon as it wears off they walk right into your weapon.

ARCHERS- If you're skillful enough, you might be able to work your way between an archer and an enemy like this:

ARCHER

YOU

ENEMY

If the archer fires, dodge the arrow(s) and with a bit of luck, he might hit or kill his companion.

BUG- I can't be sure if this bug will work as I haven't tried it myself, I've only heard about it. If anyone has done it, please let me know, (If you have a modem, leave a message for me on Amiga Limits 725-2895 1200 baud) Anyway here it is-

I have heard that at the Citadel of Doom, if you are rooly fast, you can skirt over the lava onto a ledge by the door. So that means you don't need the rose to cross the lava, which means you don't need the 5 statues to get into the hidden city, which means you don't have to kill the witch, figure out the

swan, or even find the sorceress on the Isle of Sorcery. All you need is the wand to destroy the Necromancer, the bone and the shard.

WAND- The wand is with the dragon in the Ice Mountains.

BONE- The bone is in a room in Hemsath's tomb.

SHARD- Go with the bone to the crypt into the graveyard at midnight. Get the shard in return for the bone.

That's all for now!!!

=====================================

Dungeon Master.
A Review By Peter Ward.

My first thoughts on how to review this game, were to simply say, GO OUT AND BUY A COPY! The game, in terms of its playing style is reminiscent of "Faery Tale" in that there is alot of territory to explore, and you must decide what to do in "real-time", or else the baddies will, as the Famous Bluebottle puts it.. "deaded you".

The package consists of a not so short story and playing instructions, and one copy protected disk (which Marauder II version 10 will not crack...sigh). You are told about a powerful wizard who, in short, played with fire, blew himself into limbo, put the world at risk, and asked his apprentice to fix the mess.

To start the game you, surprise, enter a Dungeon, which is depicted in graphics that are of a standard similar to that found in Cinemaware. Game play is directed via the mouse, which will take you down dark corridors with sometimes eerie digitised stereo sounds, to the Hall of Champions. In The Hall, you have the option of resurrecting or re-incarnation of your band of up to four Champions from a group of twenty four. Each Champion has various attributes, be that as a fighter, wizard, ninja or priest which should be considered before they are bought back to life.

Fighters, use the heavy weaponry, and are generally the physically stronger of the four. Ninjas use a more "surgical" approach to doing a baddie in, and are dexterous. Wizards muster magic forces for battle and Priests are good for healing potions after battle. Your Champions also have varying degrees of health, stamina and a thing called manna, the latter being rather much like "The force being with you".

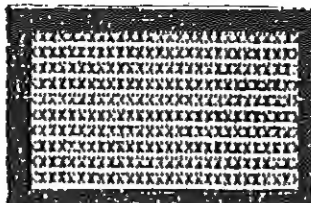Apart from the ability to slash with a sword or chop with an axe, your champions can also cast magic spells. Casting a spell to quote the instructions, requires skill (or Wisdom at least) and practice. They were not kidding, as due to the way in which spells are cast, there are over one thousand different ways of putting a hex on the baddies, or mixing up a healing potion, or whatever. The more practice you can get at spell casting, the more likely that the outcome will be favorable. The more difficult spells may be cast by the more practiced spell caster as well.

The object of the game by the way, is to recover a thing called the Firestaff, and restore balance to a ravaged world, sounds simple enough eh? So, with this in mind, down the corridors you go. As you go you will find, pieces of food and sources of water, the occasional weapon to pick up, keys to open doors, puzzles to solve and baddies to do in. One in particular is an extremely "noisy mushroom", who, as it turns out, is also edible.

I have not seen this game on the shelves in Sydney or Melbourne as yet, but a price tag of $US32.00 should give you a guide. I had no compatibility problems with my USA aquired copy. So what do I really think of this game? In short, a splendid time is guaranteed for all.....

INS OUTS & THRUS
By Neil Rutledge

Or "An Attempt To Give An Overview Of The MIDI Enviroment".

I've tried where possible to write this article with the novice in mind and at the same time give enough detail to keep the "experts" interested. This may mean the novice becomes confused and the expert bored. To the novices I say come to SMAUG (the music special interest group at the AUG) and we will answer your questions. To the "experts", come to SMAUG and answer my questions.

WHAT IS IT?

MIDI is a digital communications system which allows the transfer of information between different musical devices and instruments. MIDI, an international standard, is not audio information, rather a digital interface between musical devices that allows the controlling of "slave" devices from "Master controllers".

MIDI messages are transmitted between interconnected devices at a rate of 31.25K bits/second in a serial format (one after the other). MIDI messages are encoded into 16 separate channels which allows you to send data to a specific device on a specific channel whist sending different data on a different channel to a different device. It is this interconnection of devices, associated hardware and software that makes up the MIDI environment.

INTERCONNECTION OF DEVICES/INSTRUMENTS

The midi environment contains master and slave equipment. Master control equipment generally generates clock information which is transmitted to the slave equipment. The slave devices receive MIDI messages from the master, the messages containing data relevant to the channel being used, timing, and effects (what,which, how much & how long).

Data is transferred from one device to another via the MIDI in, out and thru ports. The in port of a slave receives data from an out port of a master. In a typical set up a master controller (in my case the AMIGA) would transmit data via its out port to the in port of a slave device. But that's where the story would finish if it wasn't for the thru port because there is no connection between the in and out ports within a MIDI device. The in, naturally enough, goes in to be processed by the device, be it to play notes or start a drum machine. The out port is

transmitting the data generated by the device, the notes you are playing.

The data that is received by the in port is processed by the device but it is also (before the processing) re-transmitted out the thru port. This now allows for the connection of a second slave with the thru of the first slave connected to the in of the second slave. This chaining can go on infinitely. However in practice there are limitations, the biggest being timing delays. As discussed later there is an electrical buffer on the input of all MIDI in ports. This buffering introduces a finite transmission delay between the in and the thru ports which is multiplied when many instruments are chained together.

There is a way to alleviate this delay. It employs the use of multiple out ports from the master device. In this way each slave can be connected to a separate out port from the master which will introduce only one transmission delay (in the realms of micro seconds). An example of this is a standard MIDI interface for an AMIGA. The interface will most likely have one input port, one

thru port and three output ports. A MIDI message generated by the AMIGA software will appear on all three output ports enabling three separate MIDI instruments to receive the message at precisely the same time. The input port accepts MIDI messages from an external device and passes them into the AMIGA. The input port also passes the in messages out to the thru port.

Another type of delay could be called overload. Because the MIDI messages are sent at in serial format at 31.25 k b/s then it should be possible to generate so many messages that the delay in queuing and transmitting these messages becomes noticable. However to produce a delay that is audible you would have to generate notes in the order of 64th triplets at tempo's greater than 200 beats per minute (MIDI overload becomes possible when sending large volumes of data ie. three players playing synths all using the pitch wheel excessively).

Typically though the delays experienced in a MIDI system are not the transmission of the MIDI message but in the delay between processing of the MIDI on message and starting the tone generators in the slave equipment. Figure 1 shows where these delays occur and the typical times.
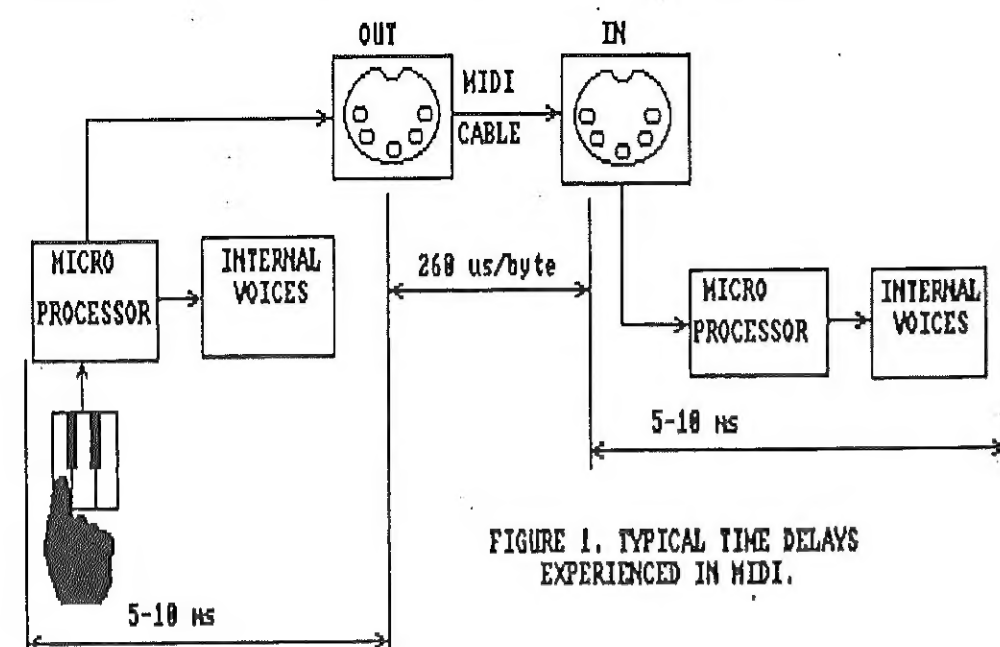

FIGURE 1. TYPICAL TIME DELAYS EXPERIENCED IN MIDI.

SO WHAT?

Well you might ask. For those who haven't become bored and gone onto the next article I will endeavour to explain how this is of use to anyone who is remotely interested in music.

If you own an electronic keyboard there is a good chance that it is MIDI compatible. Have a look at the back? (where all the plugs go) for two or three round half inch sockets which accept a plug with five pins (standard 180 degree DIN connection). These should be marked MIDI "IN", "OUT" and possibly "THRU". If they are there then your musical world is ready to take a quantum leap

forward.

The next thing you need is a MIDI interface. Why?, because all MIDI devices must be electrically isolated from each other (how and why explained later).

And finally you need some MIDI compatible software. The simplest I know of is D J Joe, a public domain real time MIDI sequencer (a sequencer is a device that plays and or records MIDI messages, it can be a seperate piece of hardware or in this case software plus the AMIGA). D J Joe is available from our Bulletin Board, Amiga Link and I'll assume that it is being used to aid simplicity.

CONNECTING IT ALL TOGETHER

Keyboard, AMIGA and interface (if it has a power switch) turned off? YES.

Using MIDI leads, (available from virtually any music shop if you don't know how to make them), connect an "OUT" port of the MIDI interface to the "IN" of the MIDI keyboard. Connect the "OUT" port of the keyboard to the "IN" port of the interface.

Connect the MIDI interface to the appropriate port on the AMIGA (most likely the serial, but there is at least one parallel interface available commercially for the AMIGA).
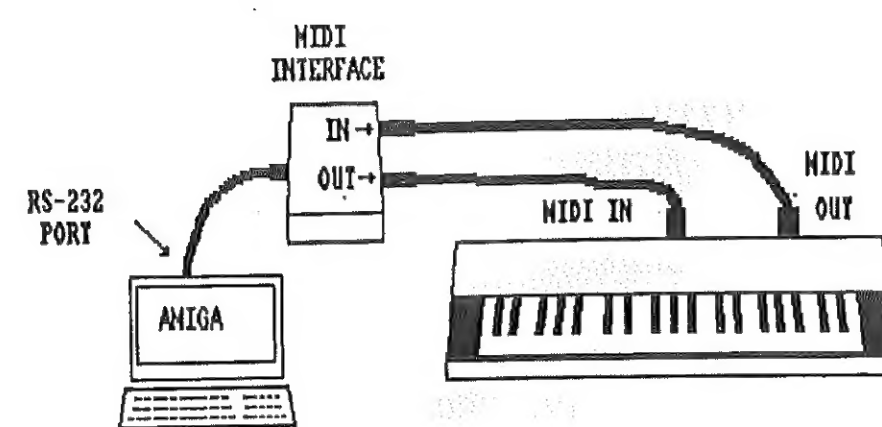

FIGURE 2. HOW TO CONNECT YOUR MIDI KEYBOARD TO THE AMIGA

Boot up your AMIGA. Turn on the keyboard (and interface if necessary). No smoke?, good.

Click on the D J Joe icon. A window will appear with a big record icon on the left and play icon on the right.

Click on the record icon (it change colour) and play a few notes on the keyboard. If everything is set up correctly D J Joe will be recording the notes

you are playing. To check, stop the recording by clicking on the record icon (colour will change back) and then click on the play icon. All being well the keyboard will play the notes you just recorded. Got the idea? Imagine the possibilities!

MIDI MODES

The MIDI system employs 16 separate channels. These channels aren't separate connections, merely designations given to the MIDI messages that allows for the filtering of individual messages on the designated channel. An example will explain this better; Two keyboards are MIDI'd to an AMIGA via a MIDI interface, the AMIGA software is sending MIDI messages to these keyboards but both keyboards are playing all the notes that are being sent. We want the treble clef to play on the first keyboard only and the bass clef to play on the second keyboard only. Simply assign a MIDI channel, ch 1, to the treble clef and a different MIDI channel, ch 2, to the bass clef in the software. Set the first keyboard to recieve MIDI messages on ch 1 and the second keyboard to recieve on ch 2.

That brings us to MIDI Modes. There are four possible modes to be considered;

    Mode 1. Omini-on, poly
    Mode 2. Omini-on, mono
    Mode 3. Omini-off, poly
    Mode 4. Omini-off, mono

Most MIDI devices (software included) can be switched between these four (not all devices offer all four modes) MIDI modes.

The term Omini-on/off describes how a MIDI device is setup to receive MIDI messages and which channels it will respond to. Omini-on means that the device will respond to incoming channel messages regardless of which channel they are received on. Omini-off is the opposite, the MIDI device will only respond to MIDI messages on its basic channel. This is normally channel 1 by default but in some cases it is possible to switch the device to receive on a different channel.

Poly, short for polyphonic, and mono describe how the notes will be played. If the device is set to xxxxx-xx, mono mode then it will only play one note at a time, that is only the last note recieved and hence will not respond to chords. In xxxxx-xx, poly mode the device will respond polyphonically (more than one sound at a time).

From the above descriptions you may have realised that mode 2 is very limited musically, what's the point of receiving notes on all channels when you

can only play one at a time.

The current use of the MIDI modes is not what the original designers of MIDI had in mind. The original specifications described mono mode as a mode where the device would respond to a number of channels simultaneously, playing one note at a time from each channel. And poly mode was to respond to one channel and do so polyphonically. A miss-understanding by the manufacture of the first MIDI keyboards resulted in the modes that we have today.

HOW DOES IT ALL WORK?

In the MIDI enviroment every note is assigned a specific value in the range 0 to 127. For example middle C on a panio is note 60. The lowest C on a panio is note number 24, and the A below it is note 21. When you play a note on a MIDI compatible instrument a "note-on" message is generated and a "note-off" message generated when the note is released.

A MIDI message is usually (there are exceptions) composed of one, two or three bytes of data sent one after the other at a rate of 31.25K bits/second. The first byte of any message is a "STATUS" byte and the following one or two bytes are "DATA" bytes. The status byte tells the processor "what", the first data byte "which" and the second data byte "how much". To make this clearer Figure 2. shows the format of a MIDI message for a "NOTE-ON" command.

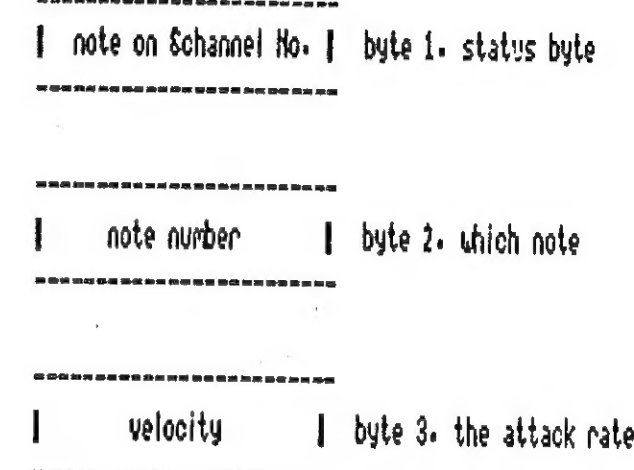| note on &channel No.	byte 1. status byte
note number	byte 2. which note
velocity	byte 3. the attack rate

Figure 3. TYPICAL MIDI MESSAGE.

THE STATUS BYTE

Breaking up the status byte gives a one bit flag to indicate that the byte is a status byte, a three bit code indicating that it is a note-on message and a four bit code to indicate which of the sixteen channels the message is in.

MSB   LSB

1 0 0 1 0 0 1 1   status byte for a note-on command

```
----------------
Bits 8 7 6 5 4 3 2 1
```

Bits 1 - 4 are used to indicate the MIDI channel, 1 - 16, being used to transmit the message. Zero numbering is used hence 0000 = ch 1, the above example is 0011 = ch 4.

Bits 5-7 define what type of message is being transmitted.

```
000  Note off command
001  Note on command
010  Polyphonic after touch
011  Control change
100  Program change
101  Channel after touch
111  System (I don't cover system messages in this
```
article)

Bit 8 is the status bit. The most significant bit (MSB) of any MIDI message is used to indicate whether the byte is a status byte or a data byte (explained next). If bit 8 is set to 1 then the byte is a status byte.

DATA BYTES FOR:
Note-On & Note-Off Commands

As explained earlier, every note has a specific number in the MIDI enviroment. When you play a note a note-on message is generated, when you release the note a note-off message is generated. The note-on message consists of the status byte which contains note-on, channel and status bits as well as two data bytes.

The first data byte contains the note number in the range 0 - 127 which is indicated by bits 1 - 7. Bit 8 is a status bit which in this case is set to 0 to indicate a data byte.

```
  MSB      LSB
  ----------------
  0 0 0 1 0 1 0 1    first data byte where x = the status bit
  ----------------
  x 0 0 1 0 1 0 1 = note No. 21
Bits 8 7 6 5 4 3 2 1
```

The second data byte contains a status bit set to 0. Bits 1 - 7 indicate the velocity with which the note was struck. If the instrument sending the MIDI message does not have a velocity sensitive keyboard a default value is sent because if no value is indicated (ie bits 1-7 0000000) then this is interpreted as minimum velocity. This would produce a note with the slowest attack rate possible.

A note-off message is very similar. The only differences are the status byte will have the note-off command (must be the same channel otherwise it will turn off a different note), and the second data byte indicates the release velocity of the note instead of the attach velocity. Once again a default is sent if the instrument does not have this facility.

Polyphonic After Touch

A Polyphonic after touch keyboard has a pressure sensor on every note, therefore there must be two data bytes sent in this message. The first contains the note number and the second a 7 bit value to define the current pressure exerted on the specific note. Status bits in both data bytes are set to 0.

Control Changes

A MIDI Control Change is a message which contains data that is not defined in any other MIDI message (simple isn't?). This data can include volume pedals, breath controllers, channel mode messages, foot switches any other knobs/buttons on the MIDI device.

The first data byte contains the 7 bit controller number and the second data byte contains the 7 bit value (some control changes use both data bytes for the "value" giving a 14 bit resolution).

Because every keyboard is different, there is no guarantee that a control change 5 on a Yamaha device is the same as that of a Casio. However most manufactures do keep some sort of similarity between their own models.

Program Changes

This command has only one data byte. It contains a status byte and a 7 bit value that represents the program change. What is a program change?. Simply, the act of changing the voice settings on a keyboard, (eg. changing from brass to strings), where the value is in this case the number assigned to the strings setting.

Channel After Touch

This is the cheap version of polyphonic after touch. There is only one sensor that gives an overall value of the current KEYBOARD pressure (not a separate value for each key). Hence there is no need for two data bytes. Only a 7 bit value which is assigned to the channel the keyboard is currently using.

Pitch Bend

Pitch bend is the act of altering the pitch of a note in small increments. The pitch bend message allows for changes of approximately 1/5th of a semitone in resolution. This is accomplished using two data bytes. The first contains a least significant byte and the second contains the most significant byte. Having two bytes sent in this manner allows for a 14 bit resolution and it is this 14 bit word that indicates how far the note has been bent.

IMPLEMENTATION CHARTS

So how do you know what MIDI events your keyboard is capable of generating and responding to ?. The answer is the MIDI implementation chart. Hopefully there will be one in your instruments manual and from it you can determine which events you have access to. The example below is almost self explanatory and is typical for a digital synthesizer.

FUNCTION		TRANSMITTED	RECOGNISED	REMARKS
CHANNEL:	Default	1	1	
	Changed	X	1-16	
MODE:	Default	mode 3	mode 3	
	Messages	X	omni on, poly, mono	
	Altered			
NOTE NUMBER:		36-96	0-127	
VELOCITY:	Note on	0	0	
	Note off	X 9nH (v=0)	X	
TOUCH:	Keys	X	X	
	Channels	0	0	
PITCH BEND:		0	0	
CONTROL	mod 1	0	0	
CHANGE:	breath 2	0	0	
	foot control 4	0	0	
	data entry 6	0	0	
	sustain 64	0	0	
	portamento 65	0	0	
PROGRAM		0-63	0-127	
CHANGE:	true #		1-32	
SYSTEM EXCLUSIVE:		0	0	data dumps, parameters
SYSTEM	Song Pos	X	X	
COMMON:	Song Sel	X	X	
	Tune	X	X	
SYSTEM				
REAL	Clock	X	X	
TIME:	Messages	X	X	
AUX:	local Control	X	X	
	All Notes off	X	0	
	Active sense	0	0	
	Reset	X	X	

```
NOTES: 0 = Supported   X = Not supported

Mode 1 OMNI ON, POLY
Mode 2 OMNI ON, MONO
Mode 3 OMNI OFF, POLY
Mode 4 OMNI OFF, MONO
```

FIGURE 4. MIDI IMPLEMENTATION CHART

ELECTRICAL CHARACTERISTICS

The principle employed in the MIDI circuitry is electrical isolation. Because different brands of equipment are going to be connected together it was decided that there would have to be some method of isolation used to prevent power, and earth loop problems.

The isolation is obtained by using an "opto-coupler" at the MIDI in port of any MIDI device. The opto-coupler acts in the same manner as the light beam found on some shop doors. An electrical signal lights the lamp, a reciver senses the light variations and converts these variations back into electrical pulses, see figure 5.
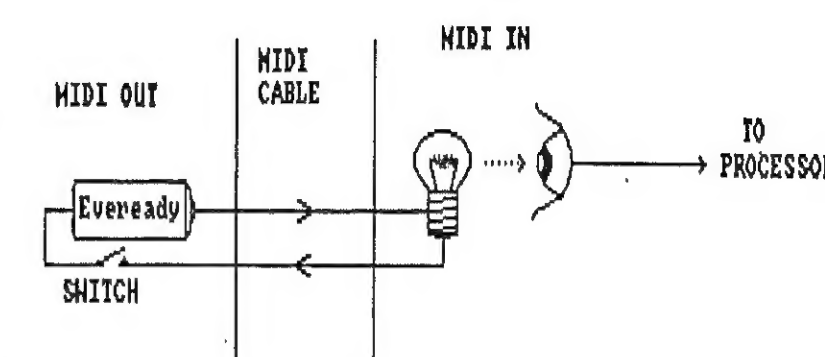
Figure 5. Basic Principle of MIDI Interface.

The MIDI system uses +5 volts as its supply reference. The MIDI data coming out of the MIDI device is low (0v) and goes high (+5v) to indicate a bit 1. However a principle called negative logic is used to reduce corruption to the data stream.

0v is a "stronger" value than +5v. That is to say a foreign potential applied to a 0v level will not have as a great an affect as would the same foreign potential applied to the +5v level. Therefore the data stream coming out of the MIDI device is

inverted (1's turned into 0's and 0's turned into 1's using the theory that there are more 0's than there are 1's).

The inverted MIDI out bit stream is then sent to the MIDI out port. At the MIDI in port the incoming bit stream is fed into the opto-coupler. This electrically isolates the MIDI out device from the MIDI in device. After the opto-coupler the MIDI in bit stream must be re-inverted before it is transmitted to the processor of the MIDI device (be it via the RS232 port or internally in a MIDI keyboard).

As explained previously the "MIDI thru" port outputs the signal entering the "MIDI in" port to enable chaining of devices. In most cases the "MIDI thru" signal is taken from the re-inverted "MIDI in" signal. This is then inverted before it is sent out the "MIDI thru" port. Figure 6 explains all this better than I can in words.
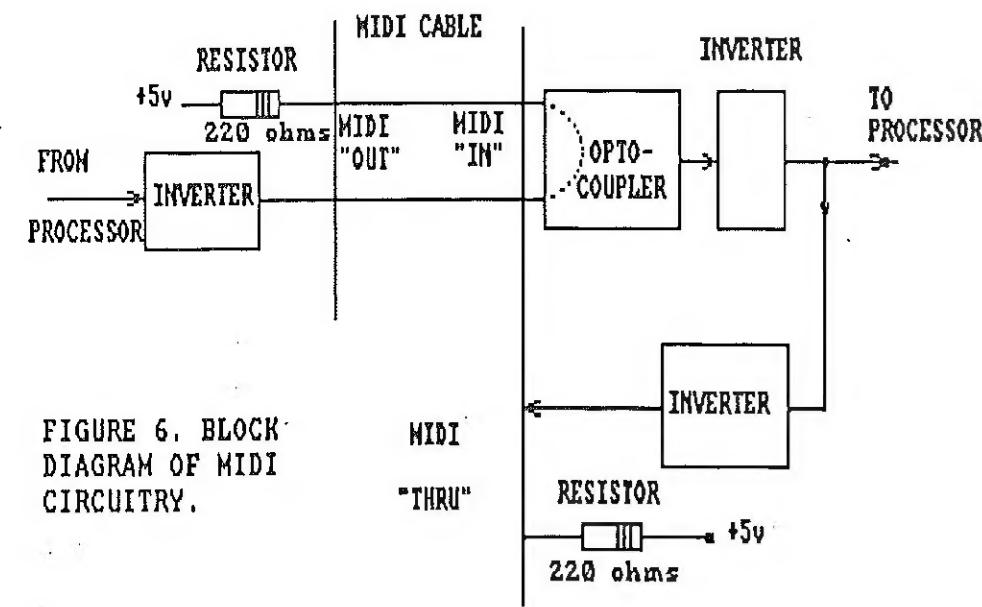

FIGURE 6. BLOCK DIAGRAM OF MIDI CIRCUITRY.

I've only just brushed the surface of MIDI and what can be done in this article. There are a lot of areas that I haven't covered such as channel mode and system exclusive messages, MIDI clock synchronization, MIDI mapping etc, but I hope I've wet your appetite. See you at SMAUG?

For further infomation there are a number of excellent music magazines on the shelves of the newsagents that contain MIDI tutorials. For any one interested in the full MIDI specification (ie. programmers writing code for an amiga based, MIDI compatible, 16 bit sampler.... hint hint?) then I point you to the;
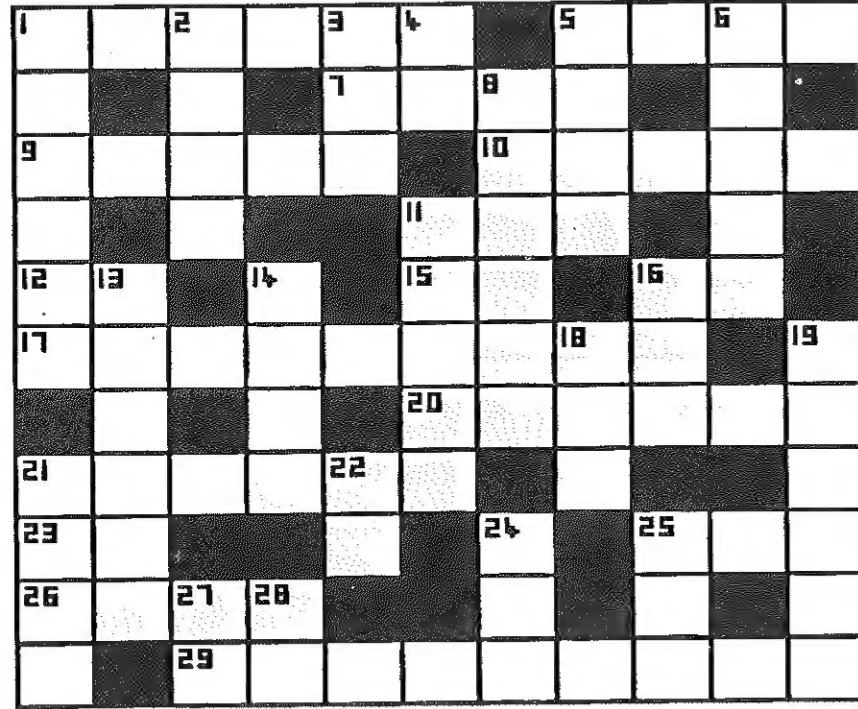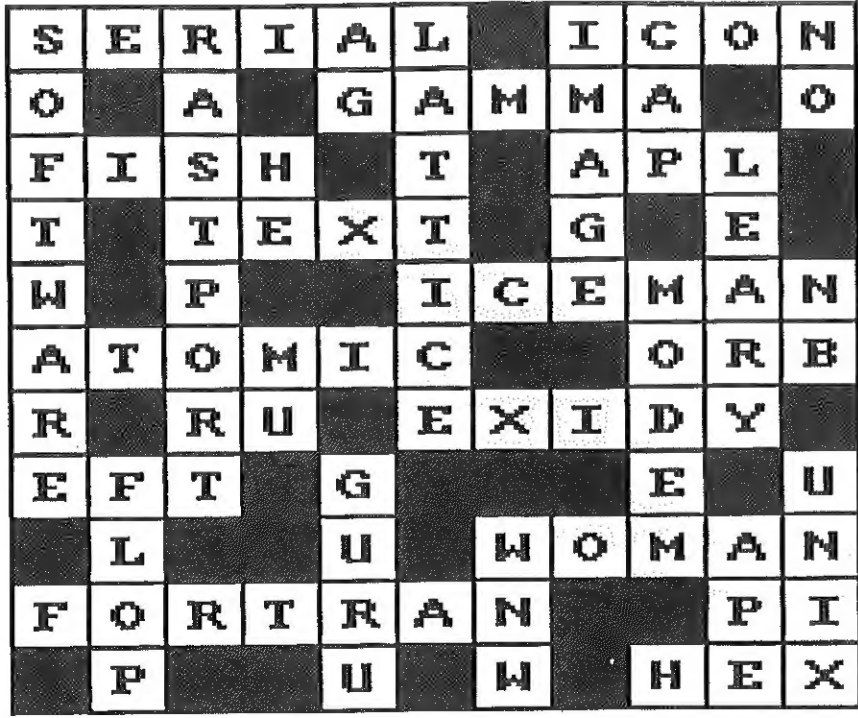
MIDI 1.0 (?)   Detailed Specification Document

If you can't find it locally, it is available from the:

International MIDI Association
11857 Hartsook St.
North Hollywood
CA, 91607

==

ACROSS

1   Clasp a programming language
2   The Gurubuster!
7   Rowing implements altered to fly
9   Used by paint programs and hairdressers
10   Let's boot out this disease
11   Just the program for an ill-mannered designer
12   Where the wizard lives (which one?)
15   Yours truly, or aluminium
16   The French
17   The company, the car, or the officer ?
20   Computerspeak for the 68000 (in our case)
21   An intuitive thingummijig ?
23   Infra-red inversion for a Tokyo mile
25   To erase, BASICally speaking
26   Data
29   The termination of a program ? No, the running of a program !

DOWN

1   (& 13D) Free software in the city square ?
2   This must be where I find music
3   Fire remains
4   Behold! it's down
5   SEC network
6   Computer rodent
8   Where Arthur was buried
11   The architect's program with the alien made the beginner
13   See 1D
14   Pollution
16   Free time wasn't sure of tropical wreath
18   Initially Robert, Graeme, and Bill produced the display
19   George, Elroy, Jane and Judy...   ...and Peter ?
21   5 was altered for a smile
22   And latin alien
24   The heart of the beast (and the brain)
25   Keyboard interface can be found in a reclining position
27   Iron
28   Beast of burden
    Apologies for 14 across last month - very esoteric !!

The crosswords printed here are courtesy of Alan Garner, who it seems has become a regular. Here is the answer to last month's crossword and a new one.

==

AMIGABasic - My First Birthday Thoughts!
by Rudy Kohut

I have been self-teaching with Amigabasic since I bought the machine one year ago. My desk now supports a large number of books on the subject, including the manual that came with the machine. This in itself may indicate, and it does, that not all that one would like to know is contained in one book. My frustrations in learning to program in this sophisticated language have been partly due to the need to piece together bits and pieces from different authors - and I still cannot find material that adequately (read 'simply') covers some of the crucial programming requirements for this language.

I would like to share with you some of my experiences, as I cannot believe that I would be the only Amiga owner to go through this process of discovery!

First, the official "manual". What a book! I have only recently begun to understand some of the jargon, and more importantly, the clues that are scattered within it (but not emphasised) which can solve many a programming problem. To be fair, the introduction does say that it assumes some familiarity with programming in Basic - how right they are! If you were a first timer like me (not having done any programming since university or college 10 years ago), the only recourse is to go to the local library and take out some very elementary books on Basic programming.

If you do that and if you think that programming might be fun after all, I would recommend you get William B. Sanders book, "The Amiga Microsoft Basic Programmer's Guide" (Scott Foresman and Company,1987). Why this book instead of, say, "AmigaBasic Inside and Out" by Rugheimer and Spanik (Abacus,1988)? Well, I have both books and find that for learning AmigaBasic from scratch, the Sanders book is more appropriate. You can go on to the other book when you have the first one under control. The second book's approach to learning is by giving you rather large pieces of program to copy and then explains them. This is rather too formidable to start off with, and the learning is not sequenced properly because of the programming requirements. On the other hand, if you want 'instant success', the latter book gives you some well written programs to copy!

While I'm talking about the 'introductory' level to AmigaBasic, let me put in a plug for Compute!'s "Kids and the Amiga" (written by E. H. Carlson, 1986). This is an excellent book for learners of all ages - don't be fooled by the title! It is short, only covers some of the more commonly used commands, but is well set out, illustrated and very understandable! I wish all computer books could be as easy to follow and learn from with enjoyment.

Once you have the basics of 'AmigaBasic' under control, and you want to stretch your skills, then get hold of the following two books: (1) "Advanced Amiga Basic" by T. R. Halfhill and C. Brannon (Compute Books, 1986); and (2) "Amiga Tricks and Tips" by Bleek, Maelger and Weltner (Abacus, 1988).

Both of these books deal extensively with matters that are treated peripherally by the earlier books mentioned - in particular the "Library" calls. Unfortunately, neither book really gives "Memory Management" a decent discussion, and my lack of understanding of this critical process has caused me many a frustrating hour! I'm sure that people who have grown up programming the C64 or equivalent is quite at home with this area but I certainly wasn't.

To give you an example of how my deep ignorance is - I was working on a database program that was cobbled together from bits and pieces of many other programs that I liked. Suddenly, upon trying to load the program I was visited with the "Out of Memory" error requestor. My first one! So what to do? The manual says that I can adjust the 'Amiga Basic data area and the system stack' with the "CLEAR" command. Alright, I changed the program to include a "CLEAR" statement at the start and tried again. Guess what? The same error message!

Reading more deeply now, I find that I need to adjust the memory size before the program loads - so a CLEAR staement inside the program itself is no use! But I don't want to have to load AmigaBasic first, execute a CLEAR in the output window and then load my program - I want it to run directly from Workbench! Keep reading - "assign the required memory with a small program that executes the CLEAR statement and then CHAINs in the larger program". Yes, it now works.

But I'm still baffled by what the "Stack" is and the "System Heap". Apparantly the size of the "Heap" is important to SOUND and WAVE statements - which is probably why my music programs keep falling over (ie. not working!).

And as for PEEKS and POKES - the manual is correct in asserting that you really have to know what you are doing to use these commands. However, the many example subroutines given in the two books use these commands, and I would like to see a book examine their use thoroughly in the context of AmigaBasic.

The "Advanced Amiga Basic" book is very valuable for its discussion of the 'event trapping' routines in AmigaBasic, as well as the use of 'Library' functions. I'm finding myself using 'Library' functions extensively now, as they can be included in "subprograms" very easily. The examples of using these routines in both books are good - although be warned that the "Advanced Amiga Basic" book was based on V1.0 of AmigaBasic, which had some problems which appear to have been fixed in V1.2 - the ability to access the "Read", "Open", etc routines in the dos.library for example. Also, I have found their use of some library calls give me error requestors - usually overcome by seeing how the second book by Abacus handled the call in similar cases (see especially the "SetSoftStyle" call).

Why access the ROM libraries at all? I initially got interested when trying to place the cursor at a particular place on the screen for graphics drawing purposes - especially trying to locate text in boxes etc. The Tricks and Tips book had the "Move" routine example - which lets you place the cursor using x,y pixel count, a great improvement over AmigaBasic "LOCATE", which deals in characters. Then I got interested in changing fonts and font styles, which you can only do through library calls. And now I'm interested in better "Menus" - why can't I use the "Right-Amiga, Command Key" sequence with my menus? Of course you can, but you need to access the libraries! And do you know how? Well, the only good example I have seen is contained in - yes, another book! See Compute!'s "Second Book of Amiga" (1988).

The use of "Requestors" to alert users and provide choices makes for 'user-friendly' programs - and their use is covered in the two books, but I would point out that the "Advanced Amiga Basic" book develops an effective but lengthy requestor
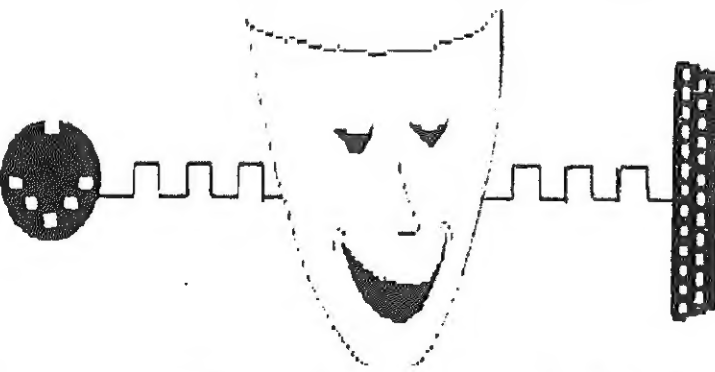
subprogram(s) that can be duplicated easily with a smaller program in the Abacus book. The advantage of the first one is that their use and the interplay of subprograms is very effectively demonstrated - so learn their method and use the subprogram from the Abacus book!

While I'm at it, the "Amiga Basic Inside and Out" book has a very good reference section at the back which explains the use of commands much better than the official manual in many cases - also the AmigaBasic error messages are listed not by number but by alphabet, so they are easier to find quickly!

The upshot of my first year learning AmigaBasic is to accept that there are many sources of information that one needs access to in order to become proficient in programming. The fact that one needs to accumulate so many different books to get an overview of the capabilities of AmigaBasic is a serious indictment of Commodore or Microsoft - probably both.
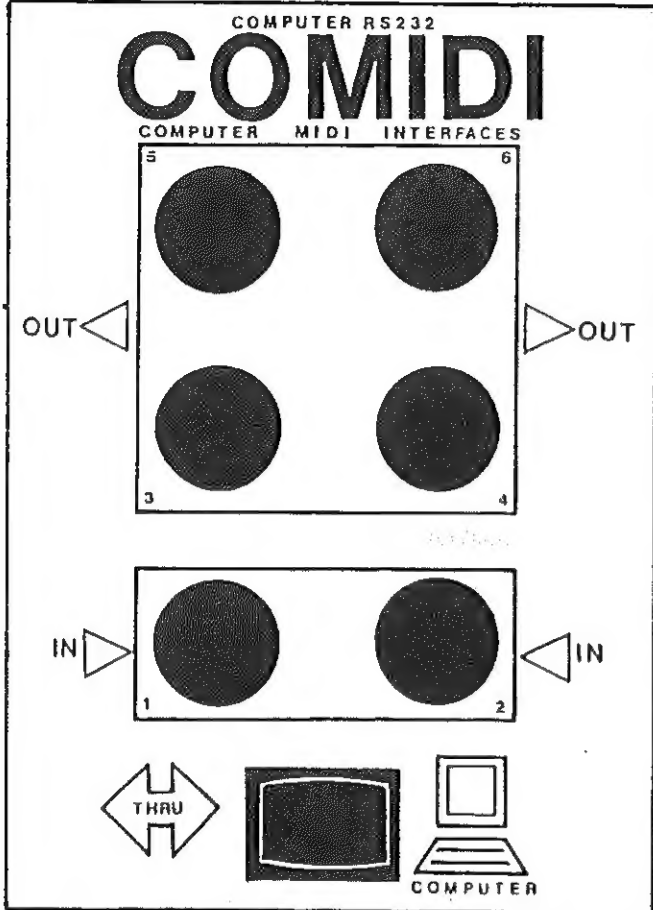
Why continue with AmigaBasic? Why not change to "C" or some other 'high level' language? My reason for sticking with it is the power combined with the ease of understanding. I am constantly amazed at what can be produced with relatively little effort. As for speed and size, that does not worry me unduly - when I become frustrated with the speed of the AmigaBasic "interpretor" I'll be able to use a "compiler", such as the one supplied by Absoft (AC Basic), which I understand is now 100% compatible with AmigaBasic. However, at my speed of learning, it will be a little while before I feel the need! And I am finding that the Library calls effectively put the power of the Amiga at the fingertips, and speed enhancements are not all that possible for many particular routines.

By the way, it is hard to find a book which gives a clear understanding of all the possible 'library' routines, especially with examples using AmigaBasic. The books mentioned above taken together give you a head start - but I had to buy yet another book to get a more complete list, "Amiga Programmer's Handbook" by E. P. Mortimore (Sybex, 1985). And that is not the up-to-date listing either as Commodore update the ROM libraries. But I have found much help in understanding the way the routines are supposed to work from this book, but at a cost of wading through programmer's jargon and having to re-read hunks.

For my next "Birthday" with AmigaBasic I hope that Commodore and Microsoft have a new version with a better and more complete manual.

==

Rocket Ranger
Game Review By Jason McConnell

With the Amiga game scene growing rapidly, more and more good quality games are being produced. And now from the makers of such classics like Defender of the Crown and Three Stooges comes Rocket Ranger, an interactive adventure set during the second world war from Cinemaware. You take the part of Cody, the 'Rocket Ranger' from Fort Dix in the U.S.A. The storyline unfolds in Chapters, the first Chapter being the kidnapping by the Germans of Proffessor Barnstorff and his beautiful daughter. You must take to the air with the help of your rocket pack situated painfully on your back! The rocket pack, if you were wondering, came from the future. You see if time keeps progressing with out the intervention of Cody, Hitler and his Third Reich will prevail and the world will be under their evil control.

You take off by pushing the fire button on your joystick each time the picture of Cody takes a step. This helps you gain speed and soon enough you are in the skies. You pick your destination using the fuel code wheel provided with game. You have to enter the right amount of lunarium to get you to your destination and back. But enough of technique. Basically what you have to do is find five parts of a rocket that are hidden all around the world in secret Nazi bases. Once you have these pieces you must find enough lunarium to get you to the Moon! The Nazi lunarium base is on the Moon you see, so you must get to the moon and destroy their base before their zeppelins bomb all of the allies out of this world.

Cinemaware have incorporated great little tunes in this game and the graphics are right up to par. In one part of the game you have to have a fist fight with a German guard to get to a piece of the rocket ship that takes you to the moon. You also get some great little pics of the proffessor's daughter. Cor!

Well that's about all of it from me. Hope to be writing further game reviews for Workbench. Bye!

==

BEATING THEM MIDI BLUES
By Justin Summers

I am writing this article for those out there, like myself, that have spent many sleepless nights trying to get their Amiga talking MIDI. The majority of my problems were not actually caused by the Amiga, but from the so called MIDI standard. The basic idea of MIDI was to allow different types of music equipment to interface with each other (i.e. Musical Instrument Digital Interface). The problems I've experienced have mostly come from assumptions made by the standard in expecting the real world to be perfect. In theory the standard has much going for it but, like all good ideas, it's implementation has left a bit to be desired.

There are two types of Byte format used in MIDI communication. The first type is known as a STATUS Byte or COMMAND byte and the second is called a DATA byte. The STATUS byte is an eight bit number that has the most significant bit set. It also contains the MIDI channel number and the command to be performed. The DATA byte is an eight bit number that has the most significant bit cleared. These contain the data to be used by the command. In this way it is easy for a device to identify commands from data and there by ignore any command and its associated data not implemented in that device. Each MIDI message consists of a single STATUS byte followed by zero or more data bytes.

The message formats fall into three general catagories. These are Channel commands, System Real Time commands, and System Exclusive commands. Its the Channel commands that have given me the most problems as these are the most commonly used commands. These include Channel Voice, Channel Mode and System Common commands.

The Channel Voice command is supposed to work in the following way. A Voice command is sent (i.e. NOTE ON command). If there is data associated with that command, (i.e. "note number" and "velocity") then this is sent after the command. That much is simple but here is where it gets a bit more complicated. The receiving device must remember the most recent command and then apply it to any subsequent data that it receives. For example if a three note cord is played on a synthesiser, one NOTE ON command and three pairs of data are transmitted (one pair of data bytes for each note in the cord). This saves the time

involved in re- transmitting the NOTE ON command.

The MIDI specification defines the Channel Voice commands NOTE ON and NOTE OFF as having two data bytes. These are the note number and velocity. This requires a total of three bytes per event (i.e. six bytes per note on/off event). In recent efforts to minimise the amount of information flowing along a MIDI channel, some manufactures have removed the velocity data byte from the NOTE OFF command (i.e. five bytes per note on/off event). To further confuse the issue the MIDI specification allows a NOTE ON command with a velocity value of zero to be used instead of a NOTE OFF command. This may not immediately appear to have reduced the amount of data but remember that once a command has been sent, the receiving device must remember it and apply it to any subsequent data it receives. Therefore after the first NOTE ON command has been transmitted only two data bytes need be transmitted for a NOTE ON or NOTE OFF event (i.e. four bytes per note on/off event).

All the sequencer programs I have tried suffer from the problem that they don't indicate the types of messages they are receiving, or indicate that they are receiving data but have not yet received a command. When starting a new track I've found that some sequencers forget the last command and will not start recording until a new command is sent. For this reason it is important to ensure that any receiving device (i.e. MIDI sequencer) has been connected to your keyboard before playing any notes. If the sequencer misses the NOTE ON command and the MIDI device does not use NOTE OFF status bytes then the sequencer will ignore all data values until the next MIDI command is sent. A way around this is to push a parameter change button or use the pitch bend wheel. This will force the synthesiser to transmit the NOTE ON command the next time a key is pressed.

Further more Midi devices will not necessarily implement the full set of commands. The MIDI specification allows a MIDI device to only partially implement the MIDI command set. This allows manufactures to produce cheap MIDI devices that only respond to a few of the commands specified in the MIDI standard.

For anyone that wants to know more about MIDI, I recommend "THE MIDI RESOURCE BOOK": by Steve de Furia and Joe Scacciaferro. This book explains the MIDI 1.0 Specification and also the

newest extensions of MIDI.

==

NWAUG NWAUG NWAUG NWAUG NWAUG	
W	U
A	North West Amiga Users Group	A
U	W
G	A Special Interest Group Of AUG	N
U	W
A	Meetings Held:	A
W	U
N	Every 2nd Wednesday	G
W	U
A	Commencing 7:30 pm	A
U	W
G	Rooms 19 & 20, 1st Floor	N
U	W
A	Essendon Community Centre	A
W	U
N	Cnr. Mt Alexander & Pascoe Vale rds.	G
W	U
A	Moonee Ponds 3039	A
U	W
G	Meetings Scheduled	N
U	W
A	1/3/89 15/3/89 29/3/89	A
W	U

NWAUG NWAUG NWAUG NWAUG NWAUG

==

Public Domain Update
by Peter Jetson

At last - more Fish disks!

Sorry for the delay in getting this lot out to you, but I had a bit of trouble finding the time to get to a bank to get a draft to send to Fred, then the draft sat here for a few weeks longer while I chased up the two faulty disks we had (153 and 155) to send back for replacement.

Anyway, here they are - disks 157 to 172.

Fish Disk #157

60or80 - A small utility to toggle the 60/80 column text modes without having to go through preferences. Works from either the CLI or the Workbench. Includes source.

AmicForm - Creates a phonebook containing only those areacodes and exchanges reachable through PC-Pursuit. Input any of Chet Solace's Finalist BBS lists and it creates the phonebook in a form usable by AmicTerm and a number of other popular terminal programs. Version 1.3, Binary only.

AnimBalls - A nifty little animation program that allows you to create a collection of balls in three-space and then interactively rotate them in real time using the mouse. Includes source.

BootBack - A handy little utility to copy and save the boot block from a disk, then later restore it should the disk get stomped on by some ugly virus. Includes source.

ECPM - A CP/M emulator for the Amiga. Emulates an 8080 along with H19 terminal emulation. Update from version on disk number 109. Includes source.

KeyFiler - A BBS message file sorter that allows sorting by keyword. Includes a textreader, Soundex matching, and limited wildcard capabilities. Version 1.0, Binary only.

ScreenZap - A little utility to clean away screens that are left by ill-behaving programs. It will kill every screen behind the WorkBench, noting how many it gets. The screens in front of WB are not affected. Includes source.

SetPrefs - Allows you to build a whole library of preference settings and instantly switch back and forth between them. Affects all preference settings not just the colors. Very useful for machines with multiple users or multiple external devices. Includes Amiga's default and various sample preference settings. Binary only. Xicon lets you use icons to call up scripts containing CLI commands. This is version 2.01, an update to the version on disk 102. Includes source.

Fish Disk #158

DiskX - Nicely done Sector-based disk editor. Binary only

MemBoardTest - Originally designed for production testing of A1000 memory boards. Very nice intuition interface. Version 2.4, Includes source in Modula.

MSDOS - A program to list files written in standard MS-Dos or Atari ST format. The files can then be copied to Ram and rewritten to disk in Amiga-Dos format. Binary only, Shareware, Version 0.1.

PCBTool - An early version of a shareware PC Board layout program Lots of options including variable size pads and traces, grids, grid snap, layers, zoom, selectable centering, text and more. This version does not support printer/plotter dumps or libraries. Version 2.6, binary only.

ScreenX - A handy little background utility that provides a small clock/memory counter in its inactive mode and a versatile screen manipulator when called upon. Binary only with source available from author, Version 2.1.

TaskX - A "real-time" task editor. Lets you list and set the priorities of all the currently running tasks. Binary only, Version 2.0.

VirusX - Update to the version on disk number 154, checks for a couple of additional new strains. Includes source, Version 1.6.

YachtC3 - Update to the Yachtc program on disk #10, contains some fixes /and incorporates a simple sound process. Version 3, includes source.

Fish Disk #159

Free - A little command to put in your c directory that returns memory status and number of tasks currently served by EXEC. Includes source.

MidiTools - A group of several different utility programs for those who run a Midi system. Binary only.

StarChart - Nicely done intuition based program to display and identify about 600 stars, galaxies and nebulae visible in the Northern hemisphere. Version 1.2, includes source.

TaskControl - Nicely done task-handling program allowing you to put to sleep, kill or change priorities of the all the currently loaded tasks. Also potentially GURU-producing, so be careful what tasks you kill, change priorities of, etc. Handy window sizer will reduce it almost to an icon to hang around until you want to use it. Binary only.

TUC - "The Ultimate Clock". Another window title clock/memory minder. This one is in 132 columns! Also gives the free memory on drives DF0, DF1 & DF2. Includes source.

Fish Disk #160

Calls - A little utility to help analyze the flow of a C-program by laying out the functions called in a hierarchical manner.

Check - A useful little utility for finding structural errors in C-source code. Many command-line options. Version 1.03, binary only.

Dis - A 68000 disassembler, written in assembly, this is an update to the version on disk #128. Includes source.

DMouse - A versatile screen & mouse blanker, auto window activator, mouse accelerator, popcli, pop window to front, push window to back, etc, widget. Version 1.09, includes source.

Update to version on disk number 145.

DWIP - "Daisy Wheel IFF Printer". A graphics printing utility that allows the printing of IFF pictures on a daisy wheel printer. Includes source.

M4 - A UNIX M4 look-alike macro processor intended as a front end for Ratfor, Pascal, and other languages that do not have a built-in macro processing capability. Pd M4 reads standard input, the processed text is written on the standard output.

MemoPad - A shareware intuition-based memo reminder program. Nicely done. Update to version on disk #146, version 1.2, binary only.

NeuralNets - A neural network example using the generalized back-propagation delta rule for learning, specifically applied to the tabula rasa Little Red Riding Hood instance.

Fish Disk #161

Friends - Cute little screen hack with command-line options to keep your mouse pointer company when you step away. Includes source.

Getsprite - A simple little program to convert Dpaint brushes into C-source. Binary only.

IncRev - A handy little program that will automatically increment the revision number of a program every time it is recompiled. Binary only.

LGZ - A Map generator/editor for the LGZ game. Not extremely useful if you don't happen to play that game, but good source example of intuition interfacing. Vesion 0.1.

Mackie - A versatile cli/macro-key initiator based on POPCLI with a unique method of "screen-blanking". I won't say more, just try it! Version 1.1, includes source.

Nag - A shareware appointment calendar with its own editor and a unique 'nagging' feature utilizing the Amiga's voice and audio devices. Version 1.6, binary only.

Perl - Practical Extraction and Report Language, an interpreted language optimized for scanning arbitrary text files, extracting information from those text files, and printing reports based on that information.

VRTest - Another anti-virus utility that allows visual inspection of ram starting a $7E7FE, ram cleaning, bootblock inspection and vector monitoring/resetting. Written entirely in assembly language. Version 3.2, binary only.

XBoot - A very simple utility to convert a boot block into an executable file so you can use your favorite debugger (Wack, Dis, etc.) to study it. Includes source.

Fish Disk #162

Avi - A workalike version of the UNIX vi editor for the amiga. Though not especially recommended for beginners, designed for those of you who may have the vi commands permanently hard-coded into your fingertips! Version 1.0, binary only.

CLI_Utilities - This directory contains several subdirectories with small utilities, collected from various sources, that are only usable from the CLI. See the Readme file for further information. Some include source.

Dark - A small graphics and animation demo. Includes source.

Flow2Troff - A little utility to convert from New Horizons Software "FLOW" files toUNIX "troff", suitable for printing on any troff-compatible laser printer. Version 1.0, includes source and a sample "FLOW" file.

LabyrinthII - A shareware role-playing text adventure game similar in operation to the Infocom text adventures. Includes source.

Iffar - Maintains archives of Interchange File Format (IFF) FORM, CAT and LIST files in a manner that complies with the IFF CAT specification. Version 1.2, includes source.

SetPALorNTSC - A couple of utility programs for testing the suitability of a developed program in either the PAL or NTSC environments. Includes source and a sample program.

TES - "The Electronic Slave" adds a gadget strip to the top of the cli window to perform such functions as device directories, info, run ED, and time. Currently, assignments are hardcoded but not difficult to change if you own a compiler. Version 1.1, includes source.

UnknownGirl - Another small musical piece similar in execution to "Synthemania" on disk number 153. Binary only.

Fish Disk #163

Bankn - A complete checkbook system offered by the author as shareware. Update to version on disk number 120. Version

1.5, binary only.

FiveInLine - A board-playing game similiar to Go-Moku, Ristinolla, etc. Fast-paced and quickly addictive! Includes source.

MachII - A "mouse accelerator" program that also includes hotkeys, the features of sun mouse, clicktofront, popcli, title bar clock with a bbs online charge accumulator, and more. Update to version on disk number 130. Version 2.4c, binary only.

MemTrace - Routines to help debug memory allocation and freeing during program development. Will complain if you try to free memory you didn't allocate and will report on memory not freed when your program finishes.

PcPatch - Patches for PCCopy and PCFormat from the 'EXTRAS 1.2' disk, to allow reading/writing/formatting 3.5 inch 360k (2 sides/ 40 Tracks/9 sectors) MS-DOS disks.

ReadmeMaster - A nifty little database for finding those programs that you know exist somewhere (???) in the AmigaLibDisk library. Maintains a key-word dictionary of the Contents descriptions that allows searching by disk number, program title, author's name, or some other descriptive word. Currently supports disks 1-154 with planned updates from the author. Binary only.

View - A mouse-oriented text file reader. Sample operation is demonstrated in reading the View.doc file, instead of using the usual "Less" text file reader.

Fish Disk #164

C-Functions - A group of four little C-functions to add to your library to make your programming life a little easier. Includes source and a small demo program showing some of the results.

DiskSalv - Very useful program to recover files from a trashed AmigaDOS disk. Can also "undelete" files deleted by mistake, so long as they have not overwritten by further disk activity. Requires two disk drives. Many enhancements since the original version on disk #20. Version 1.3, Binary only

Hed - A handy little editor that is more user-friendly than "Ed", yet doesn't require the memorization of complicated keystrokes of some of the larger, more powerful, editors. Binary only.

Newton - Uses the "Newtons Method" algorithm to estimate both real and imaginary roots of a polynomial of degree 20 or less. Version 1.0, includes source.

NewZAP - A third-generation multi-purpose file sector editing utility, from the author of FileZAP. Displays and edits full 512-byte sectors via a 106 character wide internal font. Includes a search feature to find specific strings or hex digits, forwards or backwards. Update to version on disk #58. Version 3.18, Binary only.

PcView - Provides the PC community with the opportunity to display IFF pictures to the best of EGA's ability. Displays Amiga pics, IBM-PC Deluxe Paint Pics, Apple II-GS Deluxe Paint Pics, and others in the IFF standard format. Include source.

PolyRoot - Another Polynomial root-finder using the Newtonian algorithm. Nicely done in AmigaBasic with good documentation file. Version 2.00.

PrtDrivers - A couple of new Printer Drivers. One for Digital Equipment's LN03+ laser printer, and one for Mannesmann Tally's MT420d dot matrix.

Zoo - A file archiver, much like "arc" in concept, but different in implementation and user interface details. Includes some nice features that "arc" lacks (such as file/path names up to 255 characters in length). This is version 2.00, an update to the version released on disk 136. Binary only.

Fish Disk #165

Conman - Extremely useful replacement for the standard console handler, provides line editing and command line histories. Completely transparent to any application program that uses CON: windows. This program is shareware, and well worth a donation to the author. Version 1.3, binary only, update to version on disk 133.

CPM - Another CP/M emulator independently authored from the version that appeared on disk #157. Emulates a CPM computer with a Z80 processor connected to an ADM3A terminal. Assembly source included.

Parsnag - A program to aid in performing color separations on Epson JX-80 printers. Includes source.

PlotView - A couple of programs, Plotview and Plot2Am, for viewing UNIX plot files. Also included are two sub-directories: Plot -- a device independent plotting package for the Amiga,

compatible with the UNIX plot subroutine package and Plot2Tek -- converts UNIX plot format files to Tektronix 410x terminal graphic commands. Source included.

RamCopy - A copy program designed for machines with 1 meg or more of Ram and only one disk drive. Allows you to copy a complete disk in only one pass.

SPUDclock - A simple program that uses the narrator device to speak the time at certain user specified intervals. Lots of command line options. Version 1.2, includes source.

Fish Disk #166

AutoGraf - Collects and graphically displays information on auto mileage. Features such as miles per gallon, cost per mile, miles driven, highs, lows, averages, etc. Includes sample data file, a couple utility programs and source. Version 1.0

Cref - A C cross referencer program. Prints out your code with line-numbers and complete key-word cross-referencing. Update from version on disk #103 which had a serious bug. Includes source.

MultiCalc - Yet another RPN type graphic calculator. This one generates answers with extreme precision (if 3000 digits is enough!) Features a 48-digit scrollable display, mouse driven with lots of keyboard shortcuts, and iconification.

Stevie - A public domain clone of the UNIX 'vi' editor. Supports window-sizing, arrow keys, and the help key. Version 3.10a, includes source.

Fish Disk #167

CDecl - English to C (and vice versa) translator for C declarations. This little gem will translate english such as "declare foo as pointer to function returning pointer to array 10 of pointer to long" into "long *(*(*foo)())[10]", and vice versa. Update from version on disk number 114. Includes source.

CLIcon - Allows you to run cli programs from the workbench, similar in operation to IconExec, but more versatile.

CloseMe - Another ingenious perversion in the screen hack category. Don't miss this one...surely destined to become a classic! Includes source.

DSM - (Dynamic Sound Machine) Demo version with "Save" disabled of a program that will take any IFF sound or raw data and save it as a totally self-contained, runable program.

MRPrint - A cli-based text file printing utility with lots of nice features, including tab-to-space expansion, page headers, line numbers, margin control with line-splitting and pagination correction, ARP wildcard support, and auto- rejection of files containing binary characters. Version 3.1, Includes source.

Smus3.6a - An enhanced version of the smus player that last appeared on disk number 58.

Sounddemos - Some very nice demos for showing off the incredible audio power of the Amiga!. 100% assembly language, make sure you have the stereo connected for these!

Fish Disk #168

This is the first disk of a two disk "Matt Dillion special", containing binaries and sources to the latest versions of a great number of Matt's many programs.

Fish Disk #169

This is the second disk of a two disk "Matt Dillion special", containing binaries and sources to the latest versions of a great number of Matt's many programs.

Fish Disk #170

Aftterm - Communications program utilizing IBM 3278 terminal emulation. Binary only.

Dis6502 - A ported 6502 disassembler with support added for C64 binary files. Includes source.

FastText - Blitter based fast text rendering routines written in assembly. Unique in the fact that they speed up rendering of non-proportional fonts of any height, and from 4-16 pixels in width. Source and test program included.

MRBackUp - A hard disk backup utility that does a file by file copy to standard AmigaDOS floppy disks. Includes an intuition interface and file compression. Version 2.4, Update to version on disk 129. Binary only

PtrAnim - Nifty pointer animation program, includes lots of samples, a

utility program and instructions on creating your own animations. Be sure to also read the "Disclaimer" in the author's ReadMeFirst file. Binary only, shareware

Surf - Generates bezier surfaces of revolution. Will produce some amazing pictures of wineglasses, doorknobs, or other objects one could turn on a lathe. Includes the capacity to map IFF image files onto any surface that it can draw. Source included.

Turbo - Opens a small window with a gadget that when selected, turns off bitplane, sprite, copper and audio DMA, presumably to increase system speed. Includes source.

Fish Disk #171

AZComm - Modified version of Comm 1.34 that contains Zmodem send, receive, and resume receive. Version 1.00, Binary only.

Maze - A couple of very nice demos for the creation and use of single-solution mazes, one of which is practically a stand-alone game. Includes source.

Sozobon-C - Atari ST version of what appears to be a full K&R freeware C-compiler, assembler and linker. The compiler main pass and the assembler were compiled and tested on an Amiga A2000 with only minimal changes, and they appear to work (to the extent that they believe they are running on an Atari-ST), so an Amiga port should be relatively easy.

Xoper - Very comprehensive program to monitor and control system activity. Monitor cpu, memory usage, ports, interrupts, devices. Close windows, screens, show loaded fonts or last Guru code number. Clean up memory, flush unused libraries, devices, fonts. etc. and a whole bunch more! Spawns its own process. A very handy background task to have loaded. Assembly source included.

Fish Disk #172

DataToObj - A utility to convert raw data files (sprites, image data, text, etc.) directly into object code which can then be linked to the main program without the need to go through the compiling process. Includes source.

Handshake - A full featured VT52/VT100/VT102/VT220 terminal emulator. The author has taken great pains to support the full VT102 spec. Now supports ANSI colors, screen capture and more. Update to version on disk number 60. Version 2.12a, binary only, shareware.

MFix - A small program to insert in the startup-sequence of the commercial program, Marauder II, from Discovery Software, International. Whenever the copy process is started, the rainbow screen is covered by a bare screen until the copy is finished. The author claims a 25% decrease in copy time is achieved. Version 1.0, binary only.

PopInfo - A "shrinkable" workbench utility to show you some info that Workbench doesn't, such as free memory on external devices, chip, fast, and total ram usage and more. Version 2.0, Includes source.

ProCalc - A program that simulates an HP-11C programmable calculator. Lots of enhancements and bug fixes since the original version on disk number 139. Version 1.2, binary only, shareware.

Spiff - Make controlled approximations between two files. Similar to "diff" but more versatile. Allows for the handling of numerals as string literals or numeric values with adjustable tolerances. Provides for embedded commands, scriptfiles, and many other command-line parameters. Potentially very useful, but needs some Amiga-specific work. Source and some example files included.

==
IMPROVEMENT OF AUG COMMUNICATIONS
by Bob Laidlaw

Currently, the AUG has a problem of loss of members. I have been told by some AUG members that the group could be more responsive to the needs of the newer members, who are more oriented to use rather than design of programs and hardware. There is confusion about what AUG is

doing and its future direction. Better communications between members and the AUG Committee could improve this situation.

There are several ways to improve communications within the AUG, using the three AUG communication media of Workbench, AmigaLink, and the monthly general meeting, with the objectives of increasing:

- Recognition, involvement and participation.
- Feedback between the Committee and the general membership.
- Perception that there is a reasonable balance of resources to
  the member's requirements.
- Co-operation. Dissenters can turn into contributors.

Better communications would help the Committee to find out what is being done right that should be continued, what should be changed or modified, and what should be discontinued.

Workbench:

There is a need for a greater share of space devoted to AUG as a club. Perhaps there should be less space given to articles and reviews of the kind that we can read in Transactor, AmigaWorld and Megadisk, and more emphasis to the "cutting edge" of current questions and topics of the BBS message areas.

Workbench can improve feedback between the Committee and the members, through a "Prez Sez" or "Committee Column", where we not only hear about plans for the future but also see people thanked for their contributions. And there is a need for basic information such as a list of SIG Co-ordinators' names and telephone numbers, and a Treasurer's Report each year.

AUG Monthly Meeting:

Start with a Briefing by the Co-ordinator, (from the "Briefing" agenda item at the previous Committee meeting). It is an update on "Prez Sez", so members who want the latest news, and to ask questions, have to come to the meeting. This is an opportunity to get members involved by asking for help for specific jobs, and motivate them by thanking those who respond and ask for a round of applause. Briefings are a powerful way of getting people on side, interested and informed - preventing rumours which are usually "bad news". Committee members should walk around and talk to many members to draw out ideas and feelings, and build group morale.

AmigaLink:

- Area for Committee minutes and Briefings.
- Maximise existing BBS capacity as a medium for messages.
- Put a 2400 baud modem on AmigalinkII immediately and sell the others.
- Put the most frequently downloaded files on PD disks to save member and BBS time.
- Strategic aim - one BBS in central telephone area with enough lines to serve the entire Metropolitan area and give easy access.

==

HELP

THE
AMIGA USERS GROUP
NEEDS
YOU!

AUG is setting up a user HELP-NETWORK and needs your assistance.

If you have expertise in ANY aspect of the Amiga or familiarity with commercial software packages we want to hear from you.

If you are prepared to share your knowledge with other AUG members and form part of a volunteer HELP-NETWORK please contact me to discuss how you can help support your user group.

Lester McClure
Assistant Co-ordinator AUG

A.H. 233-5664

==

Genealogy on the Amiga.

So far I have used three products for recording family histories on the Amiga. Each program has its strengths and weaknesses so I will try to outline the most important points about each.

The first is an IBM PC program called THE GENIUS. It was written by the WORDWORKS and distributed as SHAREWARE on XIDEX disks some years ago. The principle being that if you registered your copy you got a manual or if you had a hard disk you could get the extended version and a manual. The base version permits 512 records in a database where the extended (hard disk) version permits 5120 records (I believe).

Unfortunately these numbers are 'hardwired' into the program so it doesn't matter if you have a 360K, 720K or 1.44Meg floppy the program only considers storage for that fixed number of 512.

This program is the most elegant in its simplicity. On boot up you are presented with a screen showing a graphic representation of the first person placed in the database and his (meaning his or her) immediate family connected by straight lines which indicate relationships.

The selected person is in the centre (entitled "Selected"). Above left is "Mother", above right is "Father" and the horizontal bar that connects them has their marriage date on it. To the left is the "Spouse" (if any) and to the left of the spouse is a dotted line going off screen if there was another spouse. To the right is one of the siblings (if any) entitled "Brother" or "Sister" depending on gender. If there is more than one sibling a dotted line extends off the screen to the right indicating more. Below to left, centre and right are up to three of the children and a dotted line extending off screen indicating more if appropriate. The children are titled "Son" or "Daughter" depending on gender and are presented by age with eldest to the left.

The database is memory-resident (possibly a reason for the record limit). All changes made during a session are written to disk on completion of the session. Movement through the database is simple and fast. The keys on the numeric keypad are mapped to represent the people displayed on the screen. Home (7) takes you to "Mother", Up Arrow (8) takes you to "Father", Left Arrow (4) to "Spouse", Right Arrow (6) to displayed sibling, End (1) to left child, Down Arrow (2) to centre child and Pgdn (3) to right child. That person then becomes the "Selected" person and everyone on the screen is displayed relative (excuse the pun) to that selected person. One further movement is possible; F10 moves to "Show other Family", that is, the other spouse of "Selected" and children of that marriage.

Adding, editing and changing details is simple and uncomplicated. Some of the keys are mapped to carry out a function. "A" for "Add" etc. If you press any unmapped key a full screen help window pops up detailing which keys do what. F1 is Help and the help provided is always relative to the task at hand. (Context-sensitive help is the buzzword) Once a selection has been made, such as "Add", a sub-menu window pops up giving the choices of Father, Mother, Sibling, Child, Unrelated or Cancel. Each of these choices is numbered and the currently selected is highlighted in reverse so you can press the number of your choice, cursor up or down or Tab to the one you want and pressing

enter activates your choice.

In this case, Add, an empty person's details form pops onto the screen with boxes for Surname, Previous, Given1, Given2, Given3, Title, Sex, Divorced, Birth date and Death date. Only Surname and Sex are obligatory entries on this screen. You enter data and Tab to each new field finally pressing Enter (Return for A1000 users) to complete the record. Now a little window pops up on top of the entry form giving the options of Birth details, Christening details, Death Details, Burial Details or Esc(ape) to Save and Exit this entry. Choosing any of these pops up a small window over the top of the Entry window allowing you to enter Date and Place of any of these events. Each and every of these little windows can be accessed and updated until a final Esc(ape) takes you back to the main screen.

All of the facilities one would expect are there. Search, Report, Add, Edit, Connect , Break a connection, Marriage notes, Christening notes etc etc. Choosing the Search option pops up an Entry-type window into which you type any known details in any field (even parts of names) and pressing enter causes the GENIUS to display a list of up to 20 people who have those details. PgUp and PgDn allow you to scroll if there are more than can fit on the screen. Each person is numbered so typing the number or using the arrow keys or tabbing to the person you want and pressing Enter selects that person. Similarly Reports can be printed on any group of Searched For people or you can Print a report on the current Selected person.

The standard Genealogical Reports are available and are all well set out. The Pedigree Chart has the selected person and all his details on the left hand side halfway down the page, to the right of him above and below are his parents with their details, to their rights above and below each are their parents (his four grandparents) etc. All of the people are connected by lines in the usual fashion. If your printer supports compressed printing (don't they all?) the report will be printed out to 132 columns. Once the limitation of the paper is met you can print further by printing the Pedigree Charts of each of the eight Great Grandparents and their eight Great Grandparents etc.

The Descendants Chart is also well done. Along the top of the page is printed "Selected: Child: GC: GGC: GGGC: GGGGC:" and so on up to eight generations. Then the details of the P(M)atriarch are printed at the leftmost edge. Below him (her) but a couple of spaces to the right are the details of the spouse. Below them and indented to the "Child:" tab are their children and

below and indented to the next tab are their children etc. The report only goes to eight generations because of the page width limitations but the number of descendants has no limit because continuous paper can cater for any number (up to 512). Vertical stripes are printed down the page so that you can follow the indentations using the IBM DOS pipe symbol (elongated colon symbol). All of the other reports are equally well done.

Now the crunch. I feel the program is wonderful and would wish for nothing more.....Except...(that horrible word). I obtained The GENIUS when I had an IBM clone, now I don't, so I have to use the Transformer. Unfortunately the Transformer isn't perfect. It is unable to do colour on the Amiga screen so all of the windows that I have talked about each pop up in a different colour making it very clear as to which data form or menu selection is overlayed on which, but not on the Amiga. This is not particularly important but must be mentioned. The Transformer handles the DOS cursor by software because the hardware interrupt used for the cursor on the IBM is needed for other things to permit The Transformer to carry out its emulation of IBM hardware in software. Unfortunately the Transformer controlled cursor is not very reliable, it comes and goes so you don't always know which field you are in. I find myself counting the tabs and entering rubbish to see where it appears then rubbing it out, tabbing forward or back and entering the correct data. Very frustrating. Dates are even more fiddly. The other major Transformer problem is that it is slow (because of the amount of work it has to do) and saving the file on completion of a session seems to take forever.

The other limitations of the system relate to the GENIUS itself. The first is the limitation in numbers permitted on floppies and hard disk versions. The floppy limitation can be somewhat "overcome" by starting a database with each of your parents (or grandparents) being the last generation. Then you can have 512 of their antecedents and siblings per disk but you can't combine them together in a report or search through the databases collectively. The other failings are more to do with aesthetics. In a talk (demonstration) I gave to the Computer Interest Group of the Canberra Heraldry and Genealogy Society (CH&GC) the members commented that the four digit year of birth, death etc should permit a fifth character so that a question mark to indicate uncertainty about the final digit is possible. I agree with this view but the authors of this program, "Your Family Tree" and "GENP" obviously felt that no digit in any date should be entered unless the information is certain. The GENIUS and You Family Tree permit question mark substitution for

any of the digits in a date ie ??/??/???? whereas GENP uses zeroes. Finally, the CH&GC members felt that the GENIUS should permit lower as well as upper case details to be entered. The GENIUS converts everything to uppercase at input stage and thus it remains.

All in all, I like the program immensely but it is not a native Amiga program so it does not support a mouse or an Intuition environment and has the important storage limitation. In the next edition of BECAUSE I shall review "Your Family Tree" which is a US product not locally distributed and in the following month I shall give a detailed review of "GENP" which is an all Australian developed program written by Peter Evans in Melbourne. I was involved in the Beta test cycle of this product and without wishing to give too much away, let me just say that I'm leaving the best for last.

Brian Dunley

==

Scrambles
by Con Kolivas

I've decided to change the format of this scrambles article. No longer will I do the reviews, or focus on one topic for the whole article. Instead, a jumble of interesting points, rumours and tips that I think most people may find they get something out of.

First, viruses. I'm sure that by now a large number of you have heard about virusx which has reached version 3.1 (at the time of this newsletter). What is interesting is that it now checks for NINE viruses, and it doesn't even check for the HCS 4220 Anti-Virus yet. The latest virus is the first recognizable non-bootblock virus on the Amiga. It is the IRQ virus, which some bastards out there wrote (sorry, but this is how I feel about them). What it does is it checks your startup sequence and attaches itself to the end of the actual file you first ask for, or your dir command. It is harmless, but like the SCA virus can become a *major* ancestor for who knows what.

Enough of the viruses, onto NoVirus1.31 which was mentioned in last month's newsletter by Darren King. It said that it had special bootblocks to remove fast memory without destroying special bootblocks - well the programmer didn't check it on a 500! A MAJOR problem. Anyhow, I won't go into specific details, but we now have NoVirus1.49 available which has many new options and has the bugs fixed.

FFS - the fast file system. I have succesfully

mounted both my 3 1/2 inch drives using the fast file system using the program AutoDiskChange. If you don't no what I am talking about, go onto the next paragraph. What do I get out of this? MUCH faster directories, 32K more on disk, 5-7% faster reads, and 45-50% faster writes. Don't rush out and use it though, cause there isn't any checksum checking like normal AmigaDos. The reason for this is that the checking used on Hard Drives is far superior to the checksum used by the old Dos, and this system was developed for Hard Drives. I'm not sure how they are going to go about it for 1.4, but I still use FFS because it's fun. Other problems - It might access the disks about one second after what you might think it has finished with, so be careful. Any way, if you are interested, give it a bash for the fun of it, but you need the Mount from the new 1.3 commands.

Ok, now those graphics people who are even mildly interested in ray tracing have probably all heard of DBW render by David B. Wecker - I reviewed it a few months ago in this newsletter. We all know that given ample effort, it produces the best ray traced images around, so why haven't we all sent off our shareware donations? I'll give you the reason. It has finally gone commercial. However, it won't be easily recognizable as the same program. The reason for this is that (rumours here [>90% chance of being fact]) the company DBW works for wouldn't allow him to use it under his name. Therefore, he has gone all the way with the program - 3d window input and the rest - under somebody elses name! So, what is that name of the program? To tell you the truth, I haven't got the faintest idea. Being released into the public domain is a ray traced picture under Museum.arc. I've seen a small reproduction of the picture in Info,' and it looks fantastic! Under the description it explains it was produced with a modified version of DBW render (could this be the elusive program?). From what I can see in the small picture, it looks like *the* greatest ray traced image on the Amiga. It has a checkered floor (what else), a glass cylindrical case with a couple of mirror balls (of course) inside it, and some other ornaments all around the "room". Got to look out for this one!

What has happened to the Advanced Graphix SIG? Well, it seems that due to lack of turnout, it has combined with the Graphics SIG. At the last meeting it was just a friendly gathering discussing our experiences with many commercial and public domain applications. Believe it or not, I think it has made turn for the better, because the Advanced Graphix mob had trouble deciding where to draw the line between plain Graphics, and Advanced Graphics. So, now there is *no* distinction between them, we can discuss *all* topics amongst the group, and those of the Advanced Mob can help those in

the old Graphics crew, and vice versa (according to the application we are talking about). So, next month, if we manage to get a computer together, we will be showing (a couple of us, no formal demonstrations) what we have managed with certain programs, and what other programs are capable of. If you think this is your cup of tea then just come along to the SIG and check it out. I'll be showing some digitized pics, and PIXmate in particular. Who knows what others might come up with?

Postscript Laser printing. Last month and this months newsletter was produced using a PS laser, and I believe they look pretty darn good. However, I must admit that I was using Apple Hardware (the Apple LaserWriter Plus), and I'd just thought I'd share something particularly amusing about it. The Laserwriter prints at 8 pages per minute (that's fast!), and has a RAM of 1.5 Meg, for very large applications. So, why does it ONLY have serial ports? And why can they only go up to 9600 Baud? Because that's all other Apple Hardware can handle! In fact, this may not seem too bad, unless you have tried it! Certainly it prints eight pages per minute, but if you are sending (like I did) one picture - 704x650xHAM, it works out to about 950,000 bytes, and this took about 25 minutes to send to the printer! Pretty stupid I reckon. (I wonder what Commodore Lasers are like?)

Motorolla have produced a new 68030, much the same as the old one, but designed to run at 33megs! Imagine that and a 68882 running your 2000!

ASDG inc. have produced some MIGHTY software which will run with Sharps JX50 (I think that's the model) scanner. The power is in the order of: 300 dpi, 16 million colours and 256 shades of grey! However, to run it you need a 2000 with at least 2 megs and a Very Large hard disk if you want to make the most of it - works out cheaper than most other systems around! Also by ASDG, the producers of FACC, the floppy accelerator, FACC III is rumoured to be on the way out. Who knows what this one does in addition to that of FACC II?

Audiomaster II is out! I have seen messages from the programmer himself on the bulletin boards, but I haven't seen the program around here (funny, as the programmer is from Australia). Some of the features are: the ability to sample up to 56k! Yes, that's faster than the 44.1 of a standard CD player! Of course, your hardware must be able to handle this. It can put echo's and other effects on the sounds in REAL TIME and who knows what else it can do. In my opinion, Audiomaster I is still the best program around for it's job, so I can't wait to get the update!

That's enough scrambles this month. Thanks for listening.
ConMan 1.32

======================================

Editor's Column
(Written February 2, 1989)

I'm sure everybody has noticed this month's issue is a bumper issue. Well, I hope it is in a indication of articles coming in, but to be realistic, it was the fish disks this month which helped make it bigger. Anyway, thanks go to Darren King last month for the immense number of articles he produced which practically filled the newsletter. I must give belated thanks to Lester McClure here for his support and help in bringing me into this position, and printing the first few newsletters.

For this month, I've decided to tone down the virus issue, as it was taking over the newsletter, and I'm sure most of you have heard enough. Anyway, those who sent in virus articles will find their articles printed in a Virus Supplement possibly next month - as this newsletter has had enough of them.

Once again, certain things I promised some months ago didn't happen. It turns out that if we are particularly lacking on articles, I will have to do these. Otherwise, the format won't change much. A representative from Commodore in Sydney phoned me up on his own accord! He has kindly offered a copy of Professional Page and access to their laser printers if I wish it so. I haven't actually talked with him since, but I will take up the offer for the Software if it is still available - Hang in there Commodore! - recognizing their valued customers!

I have been told that the new Bulletin Board was almost impossible to find mentioned in last month's newsletter, so in a last ditch effort:

Call
AmigaLink II
(03) 376-6385
24 hours up to 1200 Baud

AmigaLink II is running a very successful trial period as our second Bulletin Board, with the NWAUG mostly catered for. It would be great to see it move up to 2400 Baud which I have found is a great topic for discussion on the Board, and requested by most. See you at the next meeting!

AUG meets on the
third Sunday of each month

Victoria College
BURWOOD CAMPUS
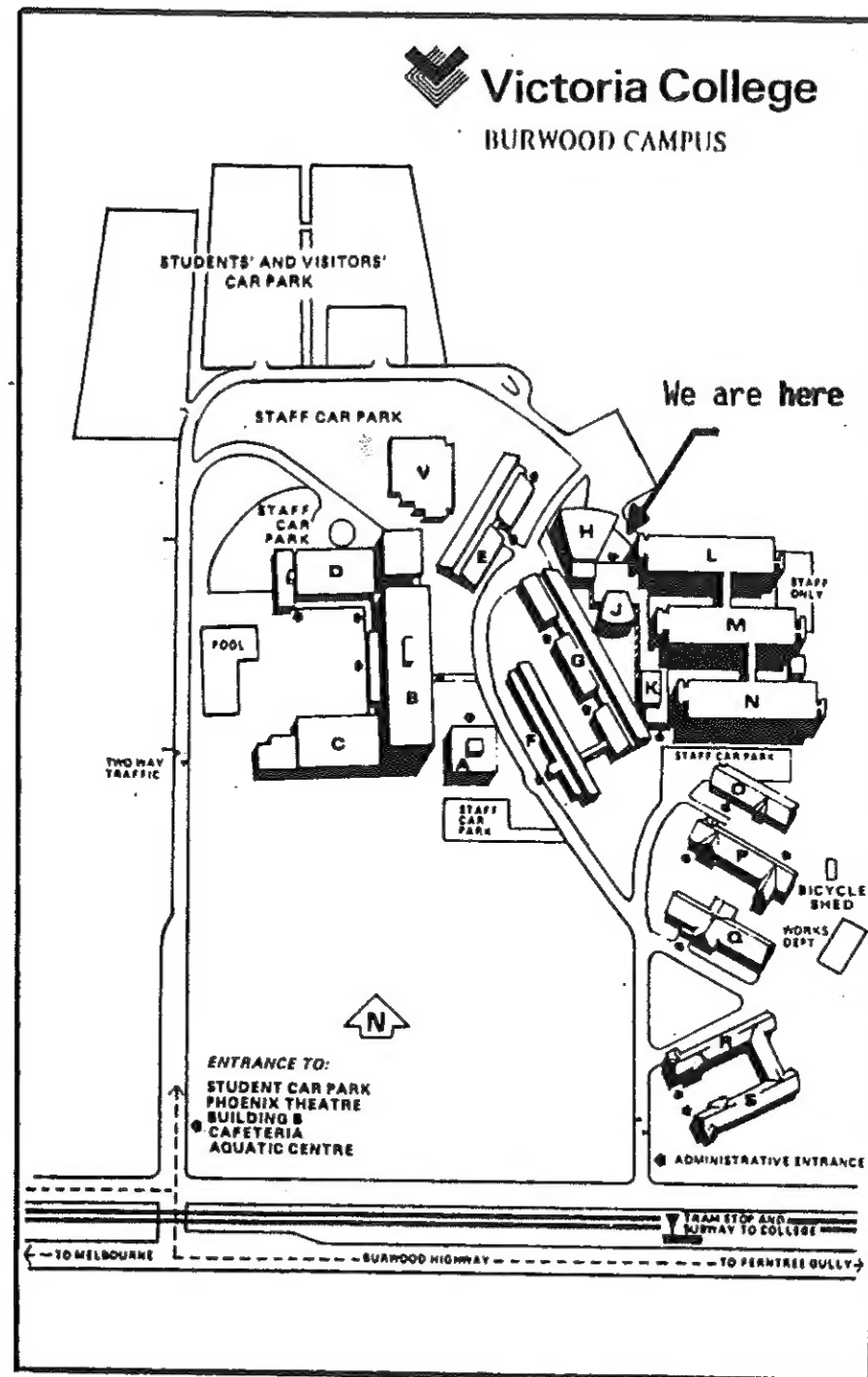
Where is Victoria College, Burwood Campus?

People often have difficulty locating our meeting place the first few times. Victoria College is on the North side of Burwood Highway, Burwood, just East of Elgar road. Coming from the City along Burwood Highway, turn left at the first set of traffic lights after Elgar road. Follow the road around past the football oval, over three or four traffic bumps to the car parking areas near the netball courts. Further up the road, to the left, you'll find Lecture Theatres 1 and 2.

If you have a Melways, try Map 61 reference B5.